



Cooffee & CODE

2

*bouw
zelf een*



*muziek
instrument*

de Bibliotheek

Utrecht





In deze minicursus bouw je zelf een muziekinstrument! We doen dat met wat simpele **elektronica** én een klein stukje **programmeren**. Geen nood, we beginnen helemaal bij het begin en lopen er stap voor stap doorheen. Over 30 minuten speel je de sterren van de hemel!



ELEKTRONICA



30 MINUTEN



MUZIEK

Het wordt geen instrument dat je bespeelt door toetsen of snaren aan te slaan, maar één die muziek maakt door sensoren te activeren. Het doet denken aan de **theremin**, één van de eerste elektronische muziekinstrumenten, uitgevonden door Léon Theremin in 1920. Net als bij een theremin, kunnen we ons instrument straks bespelen *zonder* het fysiek aan te raken. Bovendien is het heel expressief, want de kleinste beweging die je maakt is al direct hoorbaar.

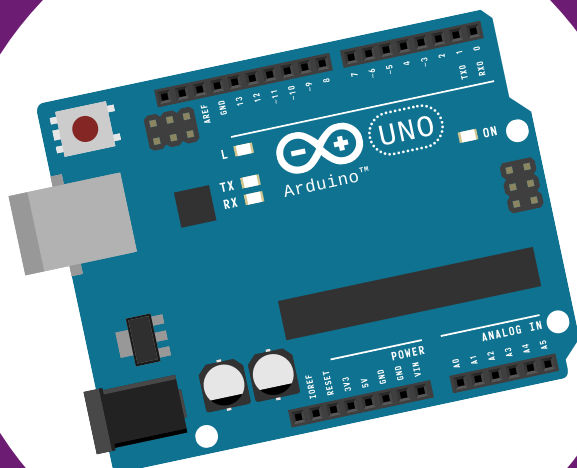


We bouwen ons instrument met elektronica, waaronder een aantal **sensoren**. Die kom je in het dagelijks leven al meer tegen dan je denkt; veel 'slimme apparaten' hebben tegenwoordig namelijk allerlei sensoren. Daarmee 'lezen' ze hun omgeving, om er op te kunnen reageren. Goed idee dus, om ook zelf eens te kijken hoe dat werkt. Zo begrijp je beter wat er om je heen gebeurt.

Ook gaan we **programmeren**. Dat is niets anders dan het opschrijven van instructies, zodat ons instrument weet wat het precies moet doen. Vrijwel elk apparaat om ons heen bevat programmeercode, van een thermostaat tot OV-chipkaart-poortjes. Maar dat zijn vooral *functionele* toepassingen; we willen laten zien dat je met programmeren ook creatieve, of zelfs *artistieke* werken kunt maken!

Step 1

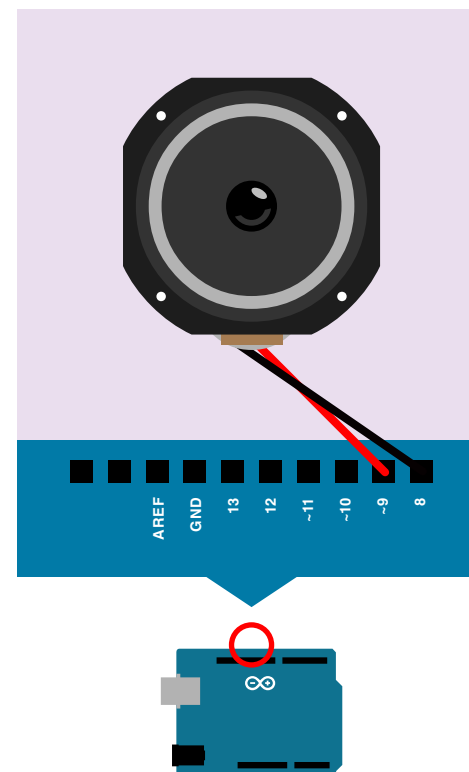
ELEKTRONICA AANSLUITEN



Er liggen al een aantal onderdelen klaar. Dit is het hart van ons instrument, de *Arduino*. Deze krijgt straks alle informatie van de sensoren binnen en zet deze om naar geluid. *Hoe* de Arduino dat doet, dat gaan we hem straks vertellen. Eerst sluiten we alles aan.

1.1

SPEAKER



Onze muziek klinkt straks uit een speaker. We sluiten daarom ook eerst zo'n speaker aan op de Arduino. Pak de speakermodule en sluit de draden als volgt aan:

● **Zwarte** draad
op nummer 8 van de Arduino

● **Rode** draad
op nummer 9 van de Arduino

TIP



Alle draadjes moeten exact op de juiste plek worden aangesloten.

1.2

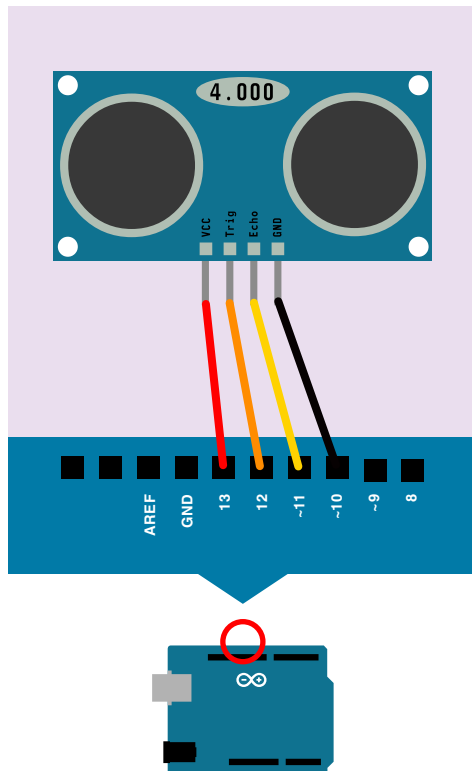
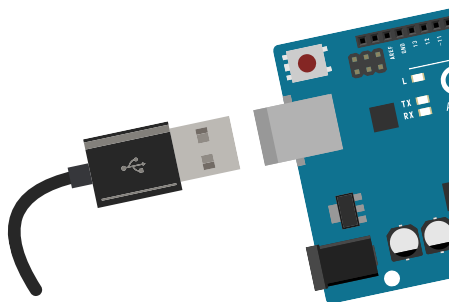
AFSTANDSSENSOR

Nu gaan we de eerste sensor aansluiten: een afstandssensor. Deze kan meten hoe ver of dichtbij iets is. De sensor stuurt ultrasone (onzichtbare) signalen uit, zoals een vleermuis doet. Als deze signalen een object raken, worden ze teruggekaatst en weer opgepikt door de sensor. Hoe verder een object van de sensor af staat, hoe langer het duurt voor het signaal weer terug is. Door die tijd heel nauwkeurig te meten, weet de sensor dus hoe ver weg iets is.

1.3 TESTEN!

Laten we het instrument alvast eens proberen. Doe de usb-kabel van de Arduino in de laptop, zodat er stroom heen gaat. Houd nu je hand boven de afstandssensor en probeer eens dichtbij en veraf. Hoor je het verschil? De afstand van jouw hand verandert nu de *toonhoogte* van het geluid.

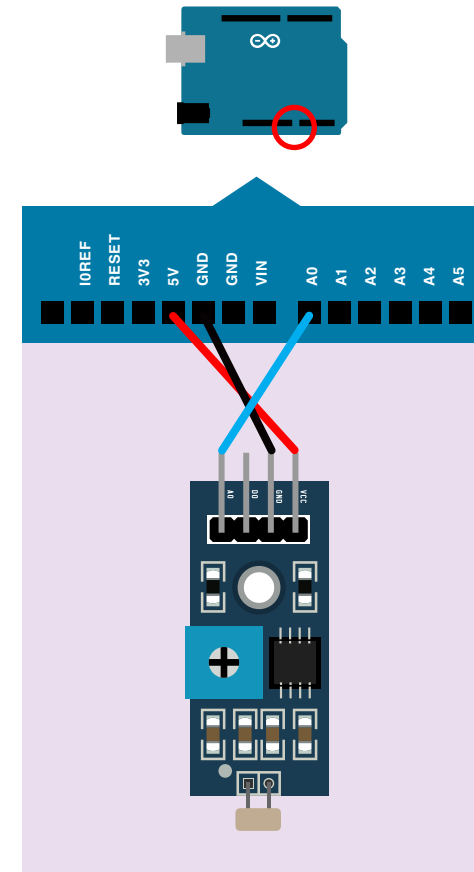
Maar; we willen ook het *volume* kunnen veranderen. Want een piano of gitaar kun je ook zachtjes spelen, of juist heel luid, dat maakt het expressiever. Dit gaan we met een tweede sensor doen. Om je beter te kunnen concentreren, kun je het geluid even stopzetten; haal het zwarte draadje van de speaker tijdelijk uit de Arduino.



- **Zwarte** draad tussen GND op de sensor en nummer 10 op de Arduino
- **Gele** draad tussen Echo op de sensor en nummer 11 op de Arduino
- **Oranje** draad tussen Trig op de sensor en nummer 12 op de Arduino
- **Rode** draad tussen VCC op de sensor en nummer 13 op de Arduino

1.4 LICHTSENSOR

De lichtsensor meet de hoeveelheid licht. We sluiten deze sensor aan op de andere kant van de Arduino, tegenover de afstandssensor:

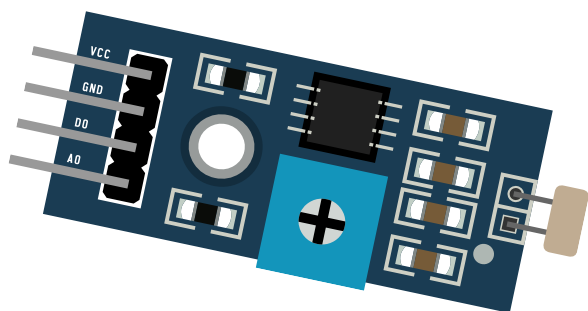


- **Zwarte** draad tussen GND op de sensor en GND op de Arduino
- **Rode** draad tussen VCC op de sensor en 5V op de Arduino
- **Blauwe** draad tussen A0 op de sensor en A0 op de Arduino

De lichtsensor werkt nog niet; eerst moeten we nog een stukje code schrijven. Kortom, we gaan programmeren!

Stap 2

LICHTSENSOR PROGRAMMEREN



We hebben net de lichtsensor aangesloten op de Arduino. Laten we eens kijken of deze het doet. We proberen de meetgegevens van de sensor in te lezen op de laptop.

2.1 METEN

Open op de laptop het bestand **theremin.ino**. De meetgegevens komen binnen met het blauwe draadje dat op A0 aangesloten zit. We vragen de Arduino daarom om de gegevens van die A0 te tonen. Zoek in de code naar de regel **// code hieronder invoeren**. Onder die regel typ je twee nieuwe regels:

```
// code hieronder invoeren
int input = analogRead(A0);
testprint(input);
```



TIP

Staat het bestand nog niet open?

Download deze dan vanaf

<http://www.setup.nl/coffeecode>

De eerste regel code vraagt om de gegevens die binnenkomen op A0. De tweede regel code zorgt ervoor dat de gegevens worden getoond. Deze code gaan we op de Arduino zetten. Klik op het pijltje links bovenin en wacht zo'n tien seconden tot de Arduino klaar is. Klik dan in het menu op **Hulpmiddelen**, en dan op **Seriële monitor**. Daar kun je precies zien welke waarde de lichtsensor meet. Probeer de sensor eens met je hand af te dekken. Zie je hoe het getal verandert?



Met deze knop zet je de code op de Arduino

2.2 SENSOR KALIBREREN

Zoals je ziet in de *Seriële monitor*, zit de waarde altijd ergens tussen 0 en 1023. Maar de lichtsensor is heel gevoelig; 0 staat voor een oogverblindend licht. Binnenshuis komt dat bijna niet voor, dus moeten we de sensor opnieuw 'kalibreren', zodat deze beter past bij onze situatie. Probeer zoveel mogelijk licht op de sensor te laten vallen en kijk dan in de *Seriële monitor* hoe laag het getal wordt. Onthoud dat getal goed! Sluit daarna de monitor weer.

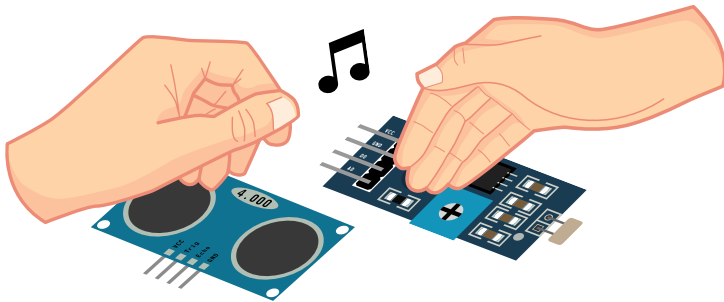
We duiken opnieuw de code in! Vul het stuk dat je net hebt geschreven aan met de volgende drie regels. Daarna vervang je de nul (0) bij `int laagste_waarde` door het getal dat je net moest onthouden. Zo stemmen we het bereik van het volume beter af op jouw omgeving.

```
// code hieronder invoeren
int input = analogRead(A0);
testprint(input);
int laagste_waarde = 0;
int hoogste_waarde = 1023;
VOLUME = map(input, laagste_waarde, hoogste_waarde, 1023, 0);
```

We sturen de code opnieuw naar de Arduino; klik weer op de knop met het pijltje (links bovenin).

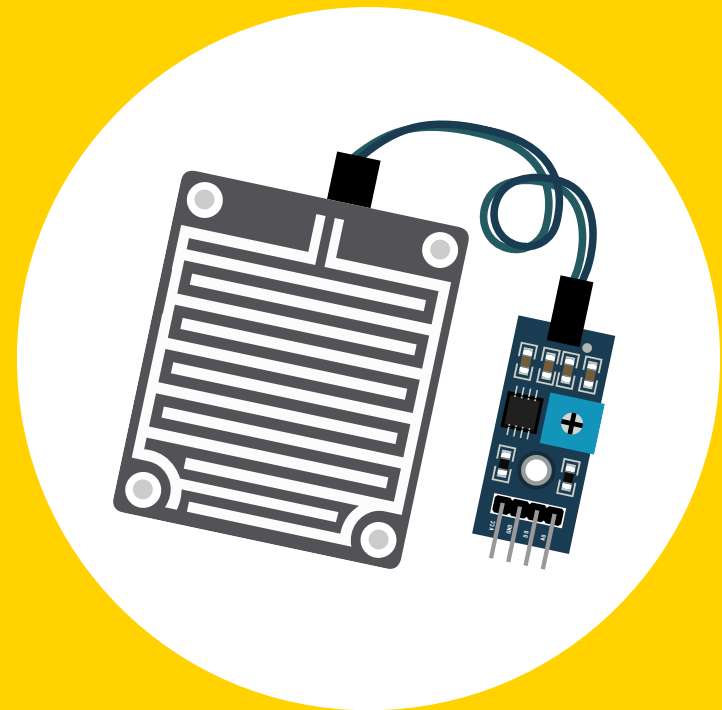
2.3 SPELEN!

Tijd om ons instrument te bespelen! Heb je de speaker weer aangesloten? Wat gebeurt er nu als je de lichtsensor (deels) afdekt met je hand, terwijl je andere hand de afstandssensor bespeelt? Hoor je hoe het volume van de muziek zachter of harder wordt? Dan is het kalibreren van de sensor gelukt! Net als een echt instrument, duurt het even voordat je het echt goed beheerst. Zin in een uitdaging? Probeer het instrument dan eens met z'n tweeën te bespelen!



Step 3

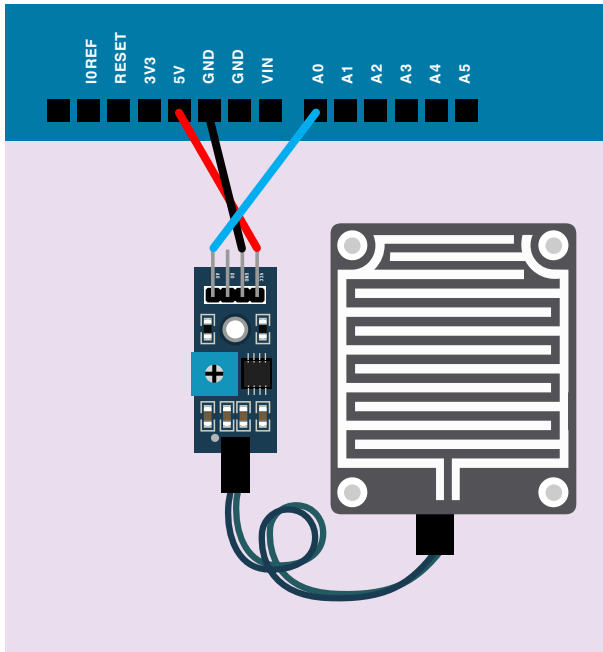
ZELF EXPERIMENTEREN



Wil je een stap verder? Laten we dan eens een andere sensor gebruiken, of zelf de toonladder van ons instrument schrijven!

3.1 VOCHTSENSOR

Wil je een stap verder? Probeer dan eens de vochtsensor aan te sluiten op de plek waar nu de lichtsensor zit. Deze sensor kun je bedienen door je vingers nat te maken, of druppels op de sensor te laten vallen. Het instellen van het volume wordt daarmee een stuk uitdagender! Je sluit deze sensor net zo aan als de lichtsensor. Met de *Seriële monitor* kun je kijken welke waarden de vochtsensor meet. Misschien wil je deze sensor ook kalibreren, net zoals je bij de lichtsensor deed?



3.2 TOONLADDER AANPASSEN

Wil je nóg een stap verder? Dan kunnen we nog de toonladder aanpassen, zodat deze meer lijkt op een piano. Zoek in de code het volgende stukje op:

```
// toonladder
const int aantal_tonen = 15;
int scale[aantal_tonen] = {
  NOTE_C3, NOTE_DS3, NOTE_F3, NOTE_G3, NOTE_AS3,
  NOTE_C4, NOTE_DS4, NOTE_F4, NOTE_G4, NOTE_AS4,
  NOTE_C5, NOTE_DS5, NOTE_F5, NOTE_G5, NOTE_AS5
};
```

Dit zijn de tonen die nu worden gebruikt. Zoals je ziet zijn het er 15. Het cijfer 3 of 4 achter de toon is om aan te geven in welk gedeelte van de piano deze noten zitten (de 'zwarte' toetsen zitten er niet bij). Om hier een C majeur ladder van te maken, pas je dit stuk code aan naar:

```
// toonladder
const int aantal_tonen = 8;
int scale[aantal_tonen] = {
  NOTE_C3, NOTE_D3, NOTE_E3, NOTE_F3,
  NOTE_G3, NOTE_A4, NOTE_B4, NOTE_C4
};
```

Upload de code nu naar de Arduino en probeer de toonladder eens uit. Hoor je het verschil? Om een stukje Vader Jacob te spelen, is de snelheid van de toonwisseling eigenlijk nog te hoog. Maar die kunnen we zelf aanpassen. In de code, boven de toonladder, staat de snelheid van de toonwisseling (**defaultDelay**, in milliseconden) en over welke afstand de 8 tonen uit onze nieuwe toonladder verdeeld worden (**maxDistance**, in centimeters).

```
const int maxDistance = 60; // in centimeters
const int defaultVolume = 500;
const int defaultDelay = 200; //in milliseconden
```



Met deze knop zet je de code weer op de Arduino

Maak het getal bij `maxDistance` nu 30, zodat de 8 noten over 30 centimeter worden verdeeld. Zet `defaultDelay` op 800, zodat de wisseling van de tonen langzamer gaat. Upload de code weer naar de Arduino, en probeer eens of je nu Vader Jacob kunt spelen:

C - D - E - C 2x

E - F - G 2x

G - A - G - F - E - C 2x

C - E - C

TIP



In het bestand **pitches.h** staan alle mogelijke tonen. Wil je een nieuwe toonladder bouwen? Klik eens op de tab **pitches.h** om te raadplegen welke tonen je kunt gebruiken!

3.3 SCHAKELN TUSSEN FUNCTIES

De lichtsensor (of vochtsensor uit stap 3.1) bepaalt het *volume* van ons instrument. Maar we kunnen die sensor in plaats van volume ook de snelheid van de toonwisseling laten bepalen. Dat doe je door in de code deze regel te zoeken, en *true* te veranderen naar *false*. Dit is eigenlijk niets meer dan een schakelaar, waarmee we tussen 2 functies kunnen schakelen. Upload de code en kijk of het instrument nu anders werkt.

```
const bool sensor2isvolume = true; // true = volume, false = snelheid
```

3.4 THUIS VERDER

Wil je thuis verder? Kijk eens op www.arduino.cc/en/Tutorial voor meer voorbeelden. Volg ook medialab SETUP (www.setup.nl) en Bibliotheek Utrecht voor meer workshops.

MEER IN DEZE REEKS



1

Creatief Programmeren
met Processing

COLOFON

SETUP.NL

Andrea Rhodenborgh

Max Laane

Frank-Jan van Lunteren

MINDMINGLE.NL

Sander van Dorsten

BIBLIOTHEEK UTRECHT

Jessie Hanssen

Coffee & Code wordt mede mogelijk
gemaakt door de Provincie Utrecht



PROVINCIE  UTRECHT



Coffee & CODE

2



ELEKTRONICA



30 MINUTEN



MUZIEK

de Bibliotheek



Utrecht

